

GUIDE TO CREATING A CSV FILE

1 We will create an Excel file.

2 Each column should be labeled based on the variable names found in the JSON keys.

Example:

invoice_id	vat_number	authorized_company	user_id	user_name	accounting_software_number	ExpRess
"195"	327185088	18	18	שם משתמש	155599981	{"Status":406,"Message":"Not Acceptable","Error_Id":8860165}

3 Each row represents one iteration an Excel file.

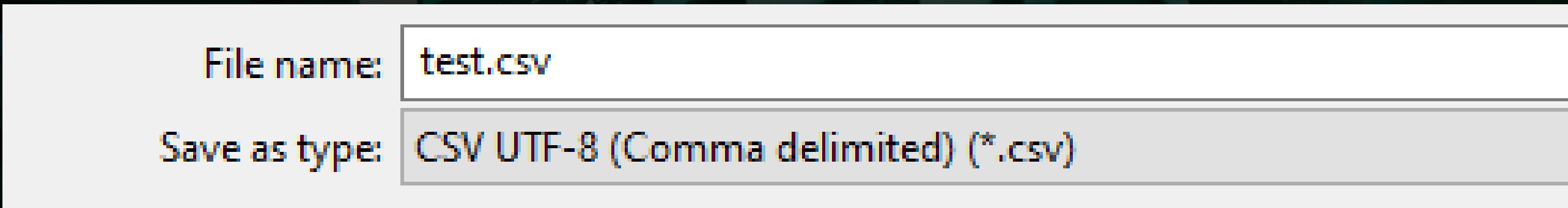
5 If the variable receives a value of type String, it should not be placed inside curly brackets, like `user_name` in the example.

4 If any variable receives a value of type number within quotation marks like a string, the value should be placed inside curly brackets in Excel, such as `invoice_id` in the example.

6 The last column should be named **ExpRess**, and in it, we will insert the expected result of that iteration.

SAVING THE FILE

Example:



- 1** Each column should be labeled based on the variable names found in the JSON keys.
- 2** Once you're done, save the file again as CSV UTF-8 (Comma delimited) (*.csv).
- 3** Close the file and ensure that you do not change anything in the file after saving it in the CSV format to maintain its integrity.

Follow the steps below to browse and select your CSV file in Excel

OPEN ANY EXCEL FILE.

FILE

OPEN

BROWSE

SELECT YOUR CSV FILE
YOU JUST SAVED BEFORE

FOLLOW THE IMAGES BELOW, CHECK THE APPROPRIATE OPTIONS, AND CLICK NEXT UNTIL YOU REACH FINISH

Text Import Wizard - Step 1 of 3



The Text Wizard has determined that your data is Delimited.

If this is correct, choose Next, or choose the data type that best describes your data.

Original data type

Choose the file type that best describes your data:

- Delimited - Characters such as commas or tabs separate each field.
- Fixed width - Fields are aligned in columns with spaces between each field.

Start import at row:

1



File origin:

65001 : Unicode (UTF-8)



My data has headers.

Preview of file F:\Fernando\אנו\אנו\בעבודה\InvoiveDecisionAPI\automation\test.csv.

```
1 invoice_id, vat_number, authorized_company, user_id, user_name, accounting_sof
2 """195""", 327185088, 18, 18, 155599981, שופר טו, {"Status": 406, "Message": "The
3 "195", 327185088, 18, 18, 155599981, שופר טו, {"Status": 406, "Message": "The
4
5
```



Cancel

< Back

Next >

Finish

Text Import Wizard - Step 2 of 3

? X

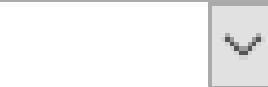
This screen lets you set the delimiters your data contains. You can see how your text is affected in the preview below.

Delimiters

- Tab
- Semicolon
- Comma
- Space
- Other:

Treat consecutive delimiters as one

Text qualifier:



Data preview

invoice_id	vat_number	authorized_company	user_id	user_name	accounting_softw
"195"	327185088	18	18	155599981	шншн шн
195	327185088	18	18	155599981	шншн шн

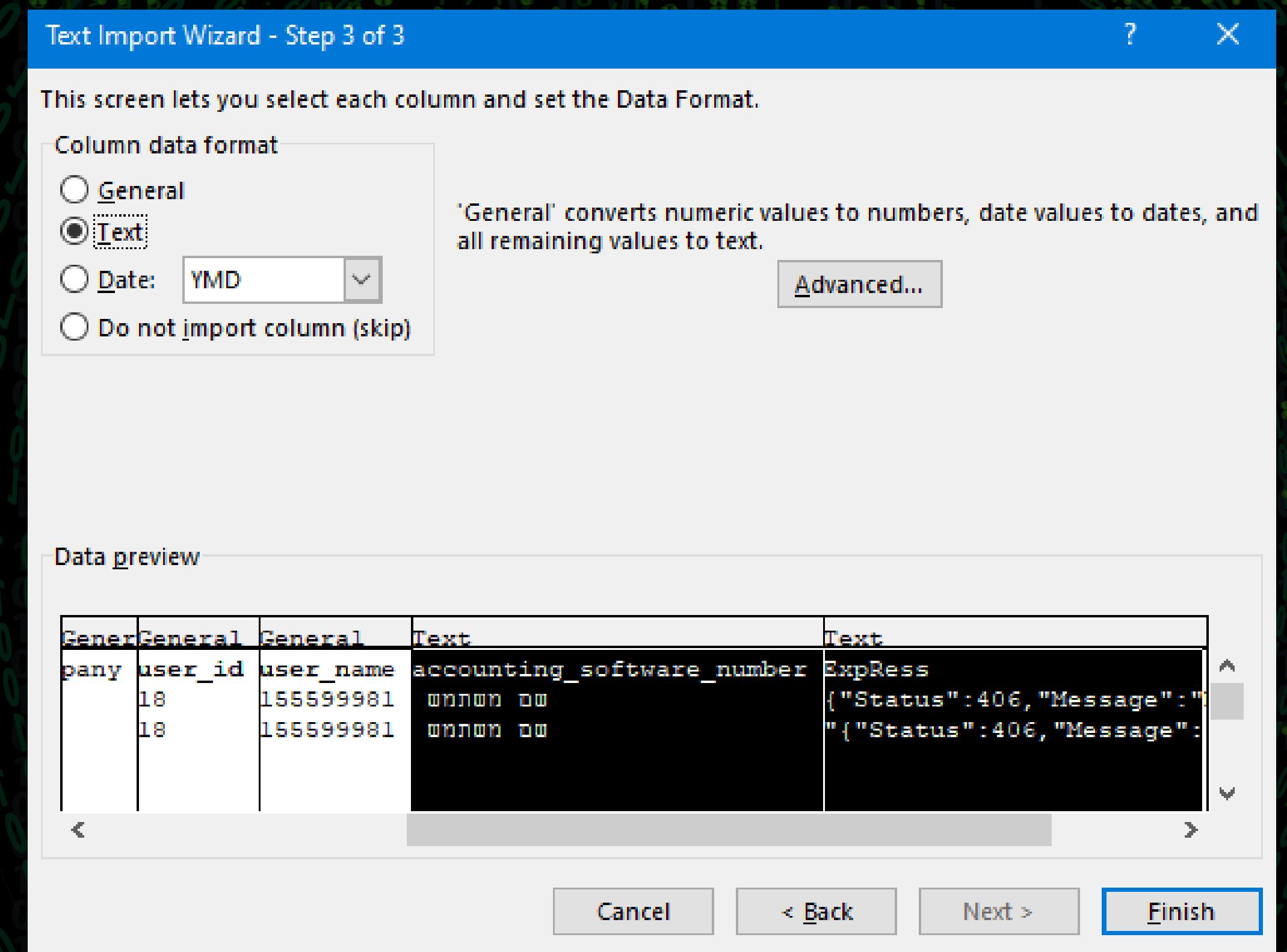
Cancel

< Back

Next >

Finish

WE WILL MARK ALL COLUMNS CONTAINING TEXT AS TEXT LIKE THIS:



AFTER FINISHING, SAVE AND CLOSE THE FILE

go to page “” and try to run if nessesity change the file continue as is

Open your CSV file using Notepad++.

OPEN
NOTE PAD

FILE

OPEN

SELECT
YOUR CSV
FILE

ONLY HERE YOU CAN MAKE
CHANGES IF NECESSARY

{"Status":406,"Message":"Not Acceptable","Error_Id":"8860165"}'". Line 3: '"195",327185088,18,18,155599981,ÿÿÿÿÿÿ, """ {"Status":406,"Message":"Not Acceptable","Error_Id":"8860165"}'"'. Line 4: '''. The background of the slide shows a blurred view of a person working at a desk with multiple monitors and a keyboard.

if there is a pattern that repeats itself, such as `"""` instead of `'', you can change it as follows:

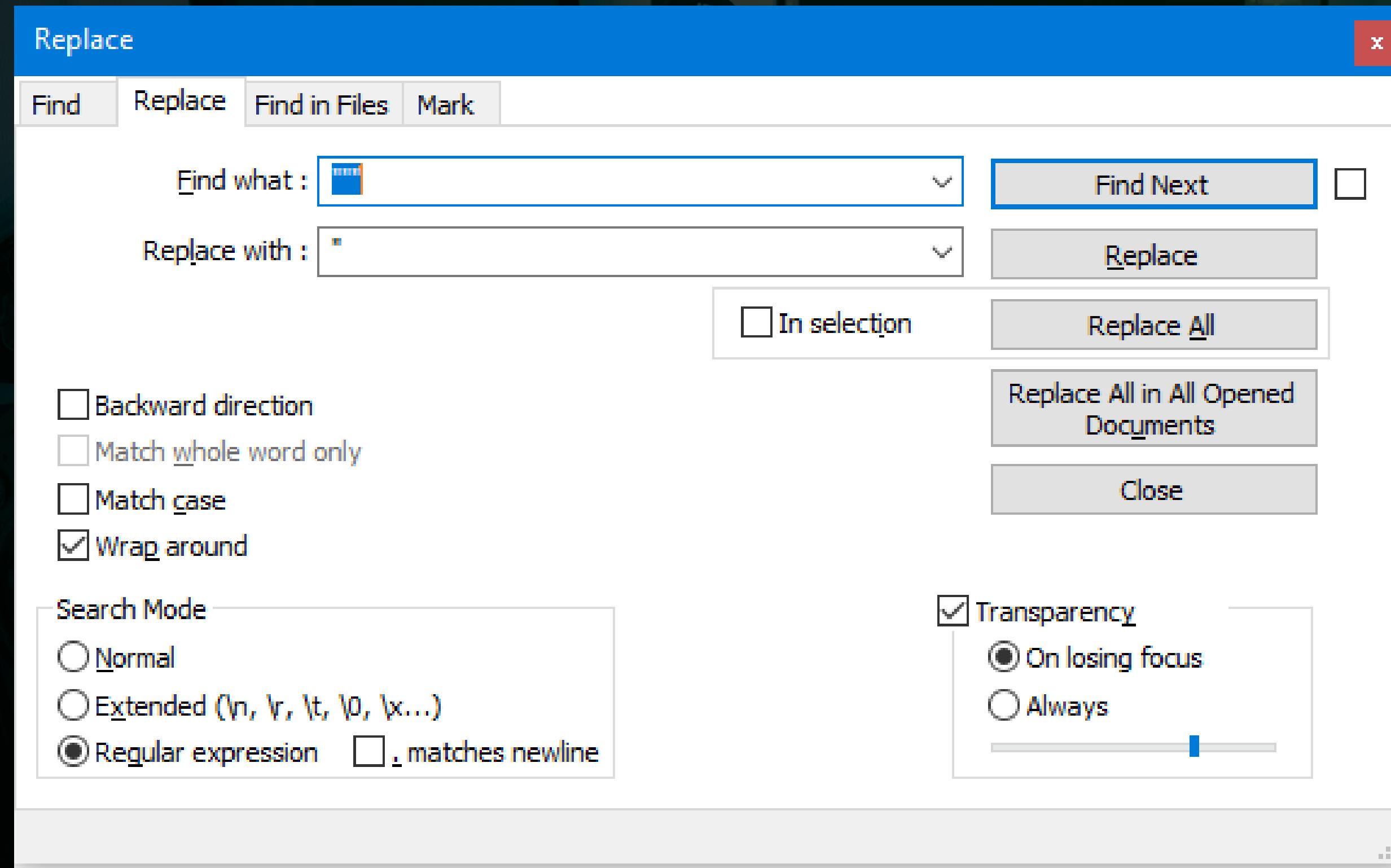
1 Highlight the pattern you want to change.

2 Press “Ctrl + F” to open the Find dialog.

3 In the “Find what” field, enter the pattern you want to replace.

4 In the “Replace with” field, enter what you want it to be changed to.

5 Click “Replace All” to replace all occurrences at once.

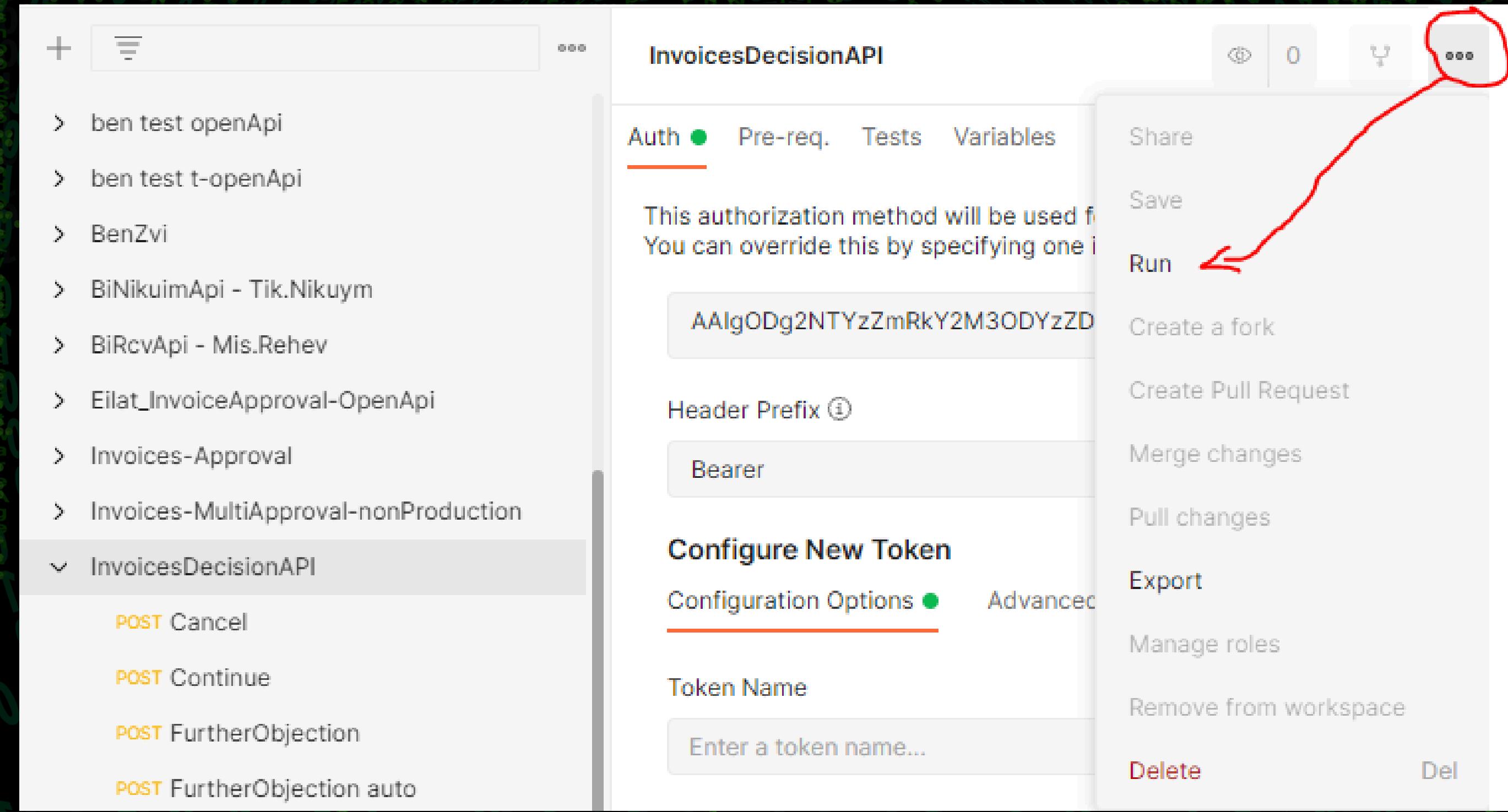


TO RUN A FILE THROUGH POSTMAN

1 Go to Collection on the left sidebar.

2 Click on the three dots (ellipses) next to the collection you want to run.

3 Select Run from the dropdown menu.

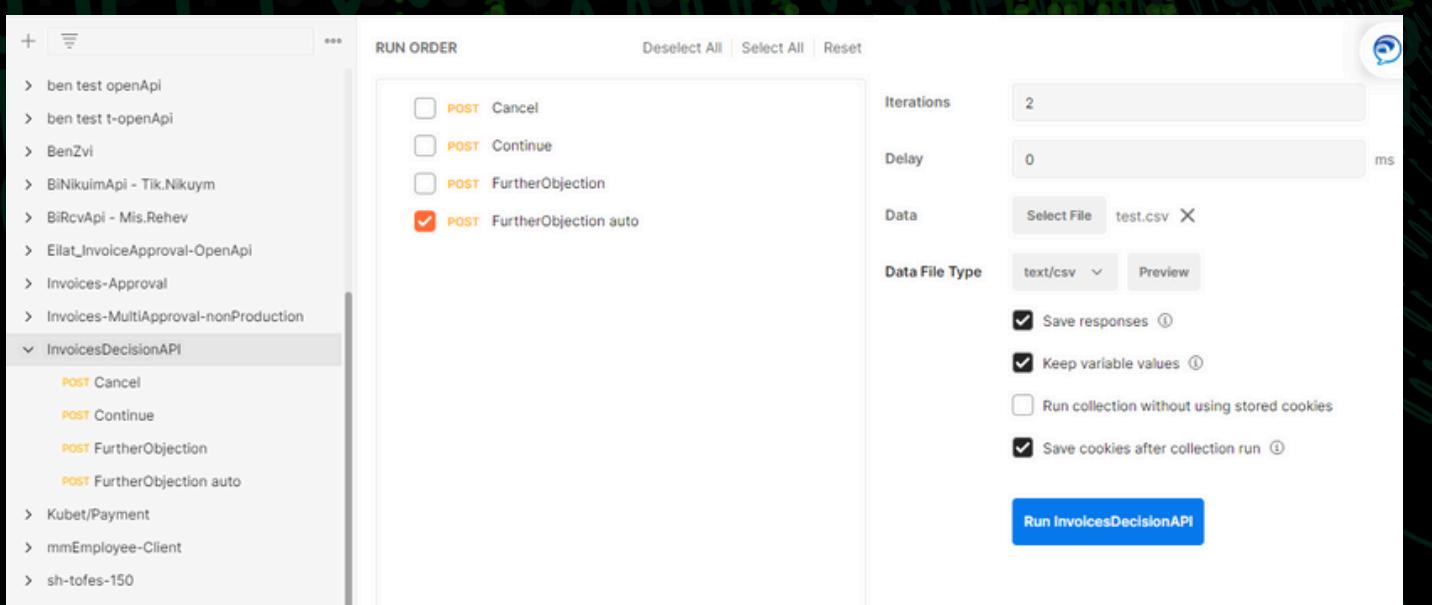


TO RUN A FILE THROUGH POSTMAN

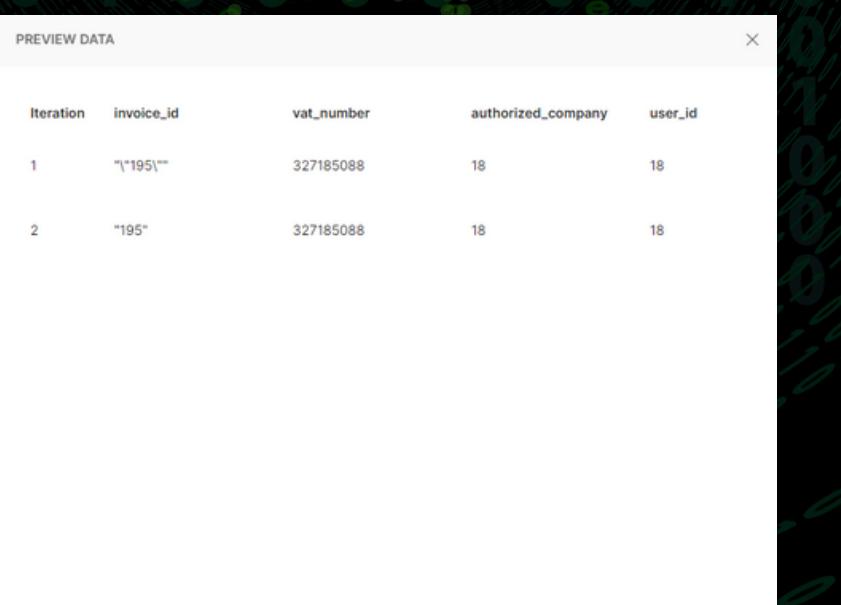
1 Select Run from the dropdown menu.

2 After selecting the CSV file, click on Preview to review the data that will be sent with each request before you run the collection.

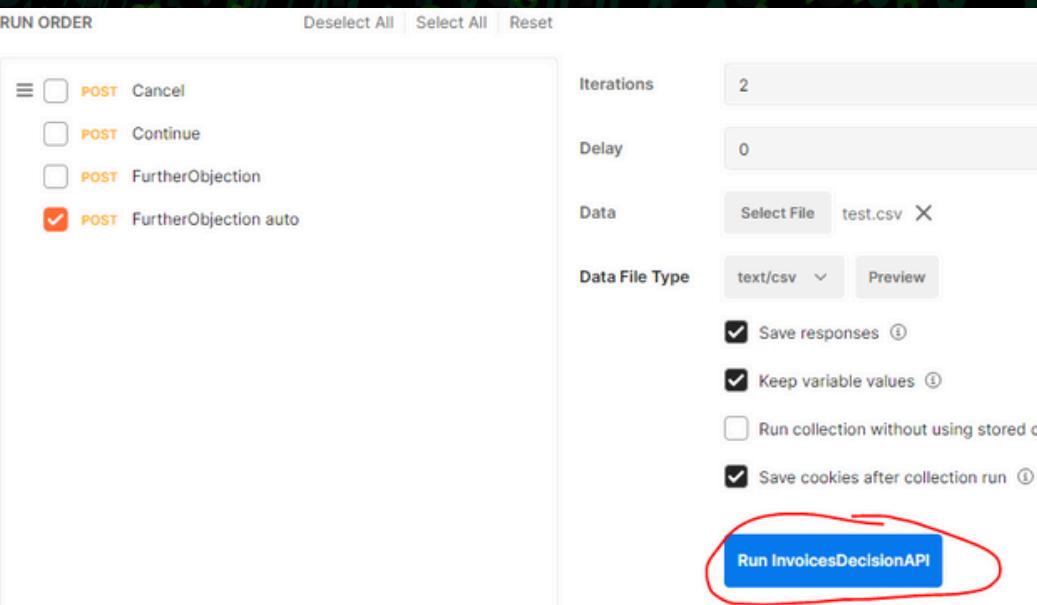
This allows you to make sure that the data is correctly configured and ready for execution. Once you're satisfied, click Run to execute the requests.



2 Check the box for Save Responses if you want to save the responses of each request.



3 Under Data, click on Select File and choose the CSV file you want to use for the requests.



TO CHECK IF THE TEST PASSED OR FAILED

1 **Results:** You will see the result of the test (pass or fail) directly in the **Collection Runner**.

This helps you identify if the test passed, and if not, you can debug by examining the request, response, and any error messages in the console.

The screenshot shows the Postman Collection Runner interface. On the left, there's a sidebar with tabs for 'Scratch Pad', 'Collections', 'APIs', 'Environments', 'Mock Servers', 'Monitors', and 'History'. Under 'Collections', several items are listed, including 'ben test openApi', 'ben test t-openApi', 'BenZvi', 'BiNikuimApi - Tik.Nikumy', 'BiRcvApi - Mis.Rehev', 'Eilat_InvoiceApproval-OpenApi', 'Invoices-Approval', 'Invoices-MultiApproval-nonProduction', and 'InvoicesDecisionAPI'. Under 'InvoicesDecisionAPI', four POST requests are listed: 'Cancel', 'Continue', 'FurtherObjection', and 'FurtherObjection auto'. The 'FurtherObjection' item is circled in red. In the main panel, a test run for 'InvoicesDecisionAPI' is shown under 'Iteration 1'. It lists four tests: 'Request URL', 'Request Headers (9)', 'Request Body', and 'Response Headers (11)'. All tests are marked as 'Pass'. Below this, under 'Iteration 2', there are two more tests: 'Response Body' and 'Ignored fields are appropriately handled', both also marked as 'Pass'. At the bottom of the screen, the 'Console' tab is selected, showing logs for the 'FurtherObjection' test. One log entry is highlighted with a red circle: 'POST https://t-openapi.taxes.gov.il/shaam/production/InvoiceDecisionApi/v1/FurtherObjection'. The log message indicates an expected result string from CSV: " {"Status":406,"Message":"Not Acceptable","Error_Id":8860165}".

2 **Detailed View:** For more detailed information:

- Open the **Console** (bottom left of the Postman window) to see logs, including requests, responses, and any test results.
- The **Console** will show you the specific details of the request and the response from the server.

3 **Request & Response:**

- Click on a specific request in the Collection Runner to view more details.
- This will show you the original request and the server's response message, allowing you to analyze what was sent and received during the test.